

Purdue University Purdue e-Pubs

Charleston Library Conference

Automating Record Loading—An Implementation Story

Maribeth Manoff
University of Tennessee

Jim Shetler
YBP Library Services

Deb Thomas
University of Tennessee, deb-thomas@utk.edu

Follow this and additional works at: <http://docs.lib.purdue.edu/charleston>

 Part of the [Library and Information Science Commons](#)

An indexed, print copy of the Proceedings is also available for purchase at: <http://www.thepress.purdue.edu/series/charleston>.

You may also be interested in the new series, Charleston Insights in Library, Archival, and Information Sciences. Find out more at: <http://www.thepress.purdue.edu/series/charleston-insights-library-archival-and-information-sciences>.

Maribeth Manoff, Jim Shetler, and Deb Thomas, "Automating Record Loading—An Implementation Story" (2012). *Proceedings of the Charleston Library Conference*.

<http://dx.doi.org/10.5703/1288284315144>

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

Automating Record Loading—An Implementation Story

Maribeth Manoff, Coordinator of Library Systems, University of Tennessee

Jim Shetler, Vice President of Library Technical Services, YBP Library Services

Deb Thomas, Research Collections Librarian, University of Tennessee

Abstract

If loading record files is consuming more and more of your library's time and effort, learn from our experience in automating record loading at the University of Tennessee. Like most libraries, the University of Tennessee Libraries (UTL) has loaded files of order and bibliographic (bib) records into our Integrated Library System (ILS) for years. In 2012, we automated this process by writing loaders that look for these record files and load them into our Ex Libris Aleph ILS without staff intervention. We began the project with Yankee Book Peddler (YBP), one of our major vendors, and the result has been positive and significant—with a few bumps in implementation. This paper covers the process from three perspectives: systems, vendor, and technical services. From a systems librarian, hear about writing the loaders—the decisions we made and factors we considered. From a vendor, see how vendors can support this kind of project, and learn a few tips to make the process easier. From a technical services librarian, learn how staff worked to implement the process, check the results, and revise our workflows. We'll share the problems we encountered, and the changes that we—systems, vendor, technical services—made together to get the project on track.

Why Did UTL Want to Automate Record Loading?

Efficiency, of course, is the easy answer—like most libraries, we are constantly challenged to do more. And like most academic libraries, we are moving away from buying print books title-by-title and into buying e-book collections. Every e-book package we buy comes with a record loading task, and every file of e-book collection records is different. Loading collection records requires staff to analyze and sometimes edit the file. We needed to automate our routine record loading so that we could move staff to the more complex task of loading records for collections. Automating routine loads meant that we wouldn't have to train more staff and that we could limit the number of people who had the permission level necessary to load record files in our ILS.

Automation also allows staff to focus their attention on parts of the process that require their expertise in acquisitions or cataloging rather than the minute details of things like filenames and dates or how to maneuver around a server on the command line. Besides making staff more productive in the use of their time and expertise, automation makes the process much less prone to error.

Why Did We Think We Were Ready to Automate?

We had at least a year of experience in manually loading most of these files, and we had manually loaded some of the files for more than 5 years. We chose to start with one vendor, YBP, so the record files were the same and did not require editing. We had established a record profile, tweaked the profile, and were happy with the records—we didn't need to review the records, just load them. We had established a regular schedule with YBP, so that our bib files were posted on Fridays, and our invoice files were posted on Saturday. A regular schedule was crucial to the success of our loaders—the bibs had to be loaded in our ILS before the invoice loader ran.

We also have a successful history of automating processes within our Aleph ILS thanks to an innovative, dedicated, and productive programmer in the Systems department, Mike Rogers. After serving as an integral team member for our Aleph implementation in 2003, Mike took responsibility for developing and implementing procedures and eventually training staff in the Technical Services unit in bibliographic record and EDI invoice loading. Loading records in Aleph

typically involves running a number of services in the GUI with options to check at each step along the way, depending on differences in the input records. After several years of refining our record customization specifications with YBP and loading records by running through these steps manually, the particulars of the records and the loading procedures were firm enough to start considering automating the process. In the meantime, Mike had developed an expertise in Aleph programming that allowed him to write custom services, shell scripts that could perform all the individual loading steps with the correct options filled in and in the correct order, from one screen in the Aleph GUI. These scripts could also run tasks such as moving files from directory to directory on the Aleph server and sending e-mails with information about the loads.

What Did We Automate?

We automated the loading of:

1. Bib records for firm-ordered e-books from ebrary, EBSCOhost, and Wiley. These bib records overlaid an order record. YBP eBook Cataloging records are highly customized to meet stringent bibliographic and local data requirements.
2. Bib records for e-preferred approvals. These full bib records created an order and encumbrance and are customized to our specifications.
3. Bib records for e-books purchased as a result of our patron-driven acquisitions (PDA) program. These full bibs create an order and an encumbrance and are customized to our local requirements.
4. Order records for firm orders of all types. These records create an order and an encumbrance. YBP calls these EOCRs (electronic order confirmation records) and they are brief, machine generated MARC records delivered in response to orders placed in GOBI, YBP's online storefront. These records are configured to convey local order data (budget codes, location codes, etc.).
5. "Discovery" records for our PDA program. No overlaying, order creation, or invoicing for this account. Discovery Records are MARC records used to underpin Demand Driven Acquisition (DDA) for digital content available from various aggregators, in this case ebrary. These records are highly customized and configured to conform to specific requirements.
6. Electronic invoices, YBP provides EDIFACT invoice support for purchases in all formats (print and digital).

Automating Record Loading from the Vendor's Perspective

The University of Tennessee is a long-standing YBP technical services customer, and their accounts produce a significant volume of metadata. For the most part, the generation and distribution of this metadata is fully automated and governed by a production schedule. The normal YBP production schedule is as follows:

1. EOCRs: Files of EOCRs are generated daily.
2. Discovery Records: Files of Discovery Records are generated weekly.
3. YBP eBook Cataloging: Files of eBook Cataloging records are generated daily or weekly.
4. Electronic invoices: Files of electronic invoices are generated daily or weekly.

The Challenge from the Vendor's Perspective

On the face of it, the generation and distribution of these various technical services products, in a way that conforms to the customer's scheduling requirements, should be a straightforward exercise. For the most part, the production work at YBP is fully automated, and it's just a matter of dovetailing the YBP production schedule with the customer's production schedule. However, this is not entirely the case because of the necessity of manual intervention, at YBP, in the generation and distribution of files of eBook Cataloging records.

At present the creation of files of YBP eBook Cataloging records is the result of a manual

process. This manual process is applied to impose a quality control check on what could be a fully automated generation and delivery mechanism. The process is as follows:

1. Worksheets are generated daily for all YBP eBook Cataloging customers. These worksheets inform YBP Library Technical Services (LTS) staff of the steps required to deliver a file that conforms to customer requirements (such as specifying an invoice date range, indicating whether or not a file must be run through customization rules and/or filters, etc.). Normally, worksheets are generated to capture the previous day's eBook invoice activity. These invoices can potentially generate at any time during a business day, and multiple invoices can be generated at different times (although this is a rare occurrence at present). As a result, the standard practice to insure that an entire day's worth of invoice activity is captured is to generate files of cataloging records for the previous day's invoices.
2. LTS staff members use the worksheets to execute "search and construct" jobs in YBP internal production systems. These jobs result in a file, or set of files, of records for each of these customers.
3. Files are reviewed to search for the existence of problems (known as "red links" and "red ISBNs" in the LTS lexicon). If no problems exist then the file is output to the appropriate FTP subdirectory, and an e-mail notification is sent to the customer to indicate that the file is available for retrieval.
4. Problems, if they exist, are routed to other LTS staff members for resolution. Such resolution usually entails intervention by a cataloger.
5. After problems are resolved, the final file construction is completed. Complete files are output to the appropriate FTP subdirectory and an e-mail notification is sent to the customer.

"Red links" and "Red ISBNs" are problems that fall into the following categories:

1. "Red links" are errors generated because of a mismatch between title data in YBP's order fulfillment and invoicing system and the metadata stored in YBP's cataloging database. Such errors include title mismatches, lack of classification data, incomplete title and subtitle data, etc.
2. "Red ISBNs" are less common errors, usually associated with discrepancies in designation and enumeration data.

Resolving these problems requires the entire daily effort of one full-time cataloger. The actual task is rotated among YBP cataloging staff members on a daily basis.

Another challenge is timing. Normally, worksheets are generated to capture the previous day's eBook invoice activity. These invoices can potentially generate at any time during a business day, and multiple invoices can be generated at different times (although this is a rare occurrence at present). As a result, the standard practice to insure that an entire day's worth of invoice activity is captured is to generate files of cataloging records for the previous day's invoices.

The service-level agreement (SLA) for the University of Tennessee Knoxville is governed by the following schedule:

File	Frequency
EOCRs	Daily
eBook Cataloging records for e-preferred approval titles	Daily (Mon-Fri)
Electronic invoices for e-preferred approval titles	Saturday
eBook Cataloging for firm-ordered eBooks	Daily (Mon-Fri)
Electronic invoices for firm-ordered eBooks.	Saturday
Discovery Records	Monday
eBook Cataloging for purchased DDA titles	Daily (Mon-Fri)
Electronic invoices for purchased DDA titles * This last invoice load on Saturday concludes with the service to find invoices with no orders	Saturday

This schedule presented a challenge for YBP, because files of eBook Cataloging records for invoices generated on a Friday were not constructed and delivered until the following Monday. This created problems for the customer, because corresponding electronic invoice files are automatically generated on Saturday. The invoice file could not be successfully processed, because a corresponding file of eBook Cataloging records wasn't available until the following Monday.

Addressing the Challenge from the Vendor's Perspective

On a macro-level, there are principles that guide this type of collaboration:

- Establish and maintain lines of communication between the stakeholders.
- Clearly define outcomes and requirements.
- Make every attempt to retain and maintain flexibility.

Designing the Loaders

Loaders had already been written for the three print book accounts we had with YBP (firm orders, approvals, and juvenile books). These loaders condensed all of the Aleph steps into one, but still required the manual step of checking the YBP FTP server for new files and uploading the right files to the right directories on the Aleph server. As the number of accounts increased to include four

different order types of e-books (firm ordered, e-preferred approval, DDA discovery, and DDA purchased), the process of keeping track of all the files on the FTP server (some of the new accounts were in different directories, and each account had a different file name prefix) became much more difficult. Mike Rogers had figured out a way to add an FTP command to the shell script to find and download files. We also envisioned that there would be no manual intervention at all for these loading processes; the loaders would be set to run automatically on a set timetable by adding them to the Aleph job list.

So we began conversations with YBP about some changes on their end that would help us accomplish this. The crucial aspects were:

1. For the script to be able to know which files to expect when.
2. That the files have a consistent file naming convention based on the date, for example, ecat32805220121015.mrc for a file of e-preferred approval bib records (our YBP number for this account is 328052) placed on the server on 10/15/12.
3. That we would receive and load the invoice files after the associated bibliographic and order records.

YBP was receptive to our proposal and we came up with this schedule:

File	Frequency	Directory	Prefix	E-mail?	Runs
YBP Gobi orders	Daily	Orders	3280	Yes	Daily @ 10 a.m.
YBP e-preferred approval bibs	Friday	ecat	ecat52	Yes	Friday @ 11:50 p.m.
YBP e-preferred invoices	Friday	invoice	3280e52	Yes	Saturday @ 1:15 a.m.
YBP firm-ordered e-bibs	occasional	ecat	ecat50	No	Daily @ 11:51 p.m.
YBP firm-ordered e-invoices	Friday	invoice	3280e50	Yes	Friday@11:52 p.m.
YBP DDA discovery bibs	Monday	discovery	ecat55	Yes	Monday @ 11:50 p.m.
YBP DDA purchased bibs	Friday	ecat	ecat55	Yes	Fridays @ 11:51 p.m.
YBP DDA invoices	Friday	invoice	3280e55	Yes	Fridays @ 11:53 p.m.

Technical Details of the Loaders

The bib and order file loaders that had already been written included the steps below:

1. Copy the files to a different directory and rename to include a date stamp.
2. Run two Aleph services to convert the file from .mrc to the Aleph loading format.
3. Run a Perl program to correct 856 fields for Table of Contents URLs to display correctly.
4. Run Aleph fix routines to make changes to the records such as:
 - Change the format of 035 fields so that OCLC number matching can work correctly.
 - Copy the 506 note provided by YBP with the number of simultaneous users to the 856 subfield z for clearer display.
5. Run an Aleph service to split the input file based on matches with existing records.
6. Run the Aleph service to load the “no match” bibliographic records.
7. Move and rename processed files.
8. Run an SQL query and PHP script to e-mail:
 - The count of the number of records loaded.
 - The order number, 245 subfield a, and order encumbrance price for each title.
9. Run a PHP program to e-mail the records that matched on an existing record.
10. Move and rename files and run 3 Aleph services to create holdings and items.

What was left to be done was to automate the manual FTP process. The new loader added this step:

1. Construct and run an FTP “Get” command based on the directory structure and file naming convention agreed to.

Once this step and the YBP schedule was in place, the running of the custom service could be added to the Aleph job list file, a list of cron jobs that run at specific times on specific days.

Automation of invoice loading was also underway before we undertook this project with YBP. There are fewer steps for invoice loading in Aleph, as in:

1. Run an Aleph service that processes all files in the edi_incoming directory and places the XML output file in the xml_incoming directory.
2. Run an Aleph service that loads all files in the xml_incoming directory.

The existing custom services for invoice loading worked like this:

1. Run the first Aleph service.
2. Run a Perl program to parse the XML output file and send an e-mail to Acquisitions staff for them to check the invoice before loading.
3. Run the second Aleph service.

The new loader would skip the manual checking step and depend on the agreed to criteria:

1. Construct and run an FTP “Get” command based on the directory structure and file naming convention.
2. Run both Aleph services, pausing in between to construct and e-mail a report with invoice information (invoice number, number of line items).
3. Set the job to run according to the approved schedule.

The last custom service we designed and implemented was one to check for invoices that didn’t have matching bib/order records. After the last invoice load of the week, this service ran an SQL query and sent an e-mail for these no match invoices.

Implementing the Loaders

We implemented the loaders March 26, 2012. In the implementation phase, we recruited staff to check the results of the record loads. We assigned one person to each account and had two basic things that we needed them to check.

First, in the early implementation phase, we wanted staff to check the results of the load: Did the loaders work correctly? Were order records overlaid, orders created, holdings generated, etc.?

When the loaders ran, they generated an e-mail containing the list of record numbers loaded in our ILS. We developed a checklist for each account, and staff used the record numbers in the e-mails to spot-check the results of the load. Overall, the loaders worked well, and Mike Rogers, the librarian who did the programming, only needed to make a few minor tweaks.

Second, we needed to check if the loader was picking up all the record files that YBP posted on their FTP site. We expected to receive a daily file or a weekly file for most of these accounts. If the designated checker didn't receive the expected e-mail message, she would ask that the FTP site be checked to look for a file. This is the area where a problem surfaced, and we needed to work with YBP to solve it.

Why didn't the loader find the record files and load them? One of these reasons:

1. The invoice file was posted before the bib/order file.
2. The files were not posted on the day the loader expected.
3. The file name was not the name expected by the loader.
4. The loader was expecting a single file for that day, but multiple files were posted. In that case, the loader picked up the first file but didn't get the others.

We discovered that the basic problem with UTL's specs for the automated process was that we based them on the schedule we had established for manual loading, which was one file per account posted on Fridays. YBP's usual practice is to post daily files of e-book records on the FTP server. Daily files permit libraries to make the e-books available to users as soon as possible. But when we were loading manually, that was too much work for us—we were loading some accounts multiple times per week. We asked YBP to create a single file for each account and post the file on Fridays, and YBP agreed to do so. This eliminated manual steps on our end, but introduced more manual intervention for YBP. UTL made the classic mistake of simply automating the current process without considering how to do things differently. It was too much work for a staff person to load files every day, but not too much work for an automated loader. The standard YBP practices would work for us if we automated the loads. We didn't need a weekly file; we needed to rewrite the loaders and let the system work.

Solving the Problem

We devised a new set of procedures built around what we thought would require the least amount of manual intervention on both sides of the equation. It was basically that YBP would post bib files for each account on the days they were completed, and those loaders would run daily, checking for a file with the given file name and running the loader when it found a file. YBP would post an invoice file for each e-book account on Saturdays. The new schedule looked like this:

File	Frequency	Directory	Prefix	E-mail?	Runs
YBP Gobi orders	Daily	Orders	3280	Yes	Daily @ 10 a.m.
YBP e-preferred approval bibs	Daily (Mon-Fri)	ecat	ecat52	Yes	Mon-Fri @ 11:42 p.m.
YBP e-preferred invoices	Saturday	invoice	3280e52	Yes	Saturday @ 11:30 p.m.
YBP firm-ordered e-bibs	Daily (Mon-Fri)	ecat	ecat50	No	Mon-Fri @ 11:44 p.m.
YBP firm-ordered e-invoices	Saturday	invoice	3280e50	Yes	Saturday @11:32 p.m.

YBP DDA discovery bibs	Monday	discovery	ecat55	Yes	Monday @ 11:45 p.m.
YBP DDA purchased bibs	Daily (Mon-Fri)	ecat	ecat55	Yes	Mon-Fri @ 11:40 p.m.
YBP DDA invoices * This last invoice load on Saturday concludes with the service to find invoices with no orders	Saturday	invoice	3280e55	Yes	Saturday @11:34 p.m.

On UTL’s part, solving the problem required a rewrite of the loaders to match the new schedule. On YBP’s part, the solution required an alteration of internal production schedules. To be specific, files of eBook Cataloging records for this customer are produced the same day as invoice generation. This alteration is the only way to insure that all files of eBook Cataloging records are made available for automated retrieval and processing before the corresponding files of electronic invoices. As a result, the following workflow now governs the creation and delivery of technical services products for this customer:

1. Files of eBook Cataloging records are produced the same day as the associated invoice. This is possible because, to date, no invoices for purchased content are generated for this customer after 3:00 AM.
2. The eBook Cataloging worksheets for this customer are the first to be worked on every business day. This enables YBP LTS staff to identify problems early in the day.
3. Any problems for this customer (“Red links” and “Red ISBNs”) are distributed for resolution immediately upon discovery.
4. Problem resolution for cataloging issues is given top priority and all problems resolved that day.

Lessons Learned and Outcomes

For YBP, the challenges associated with meeting this client’s needs clearly demonstrate the tension that exists between synchronizing fully automated

processes with the desire to deliver a high quality metadata product that is error free and that fully meets customer expectations and requirements. Unfortunately, quality control for a metadata product created by humans requires human intervention, and that intervention cannot be fully automated. Until such metadata creation is fully automated YBP staff will be racing against time, on a daily basis, to insure that products and services conform to the service-level agreement.

From UTL’s perspective, we realized that our initial problems may be a result of how we designed the procedures in the first place, meaning YBP has always been incredible and helpful and bent over backwards to accommodate our ideas about how they could change their processes to fit our needs. If UTL had it to do over again, a better approach might be to ask YBP to describe their workflow and propose some ideas of their own to fit what we were trying to do.

Both parties agree that we should have documented decisions made in conference calls so that everyone had a clear and consistent understanding.

Since we began in late March 2012, we have loaded over 2,000 records with these loaders. So far, the revised schedule has eliminated the initial problems. The files load without incident, and we no longer rigorously check the load results. The staff member who previously spent her time loading these files has been able to devote more time to loading e-book package records and is currently training cataloging staff to load e-book packages.